

CLAIM AMENDMENTS:

The following listing of claims will replace all prior versions and listings of the claims.

1. (Currently Amended) A telephone gateway device ~~telecommunication protocol engine device~~ comprising:

a first memory device storing one or more telecommunication protocol templates, each of the one or more telecommunication protocol templates including a plurality of virtual machine instructions to implement a telecommunication protocol and to implement a finite state machine;

a second memory device to store state data related to a first finite state machine;

a processor having access to the first memory device, the second memory device and firmware, the firmware adapted to operate a virtual machine, and wherein the processor:

causes the virtual machine to read first virtual machine instructions from the first memory device, the first virtual machine instructions selected from a first telecommunication protocol template to implement a first telecommunication protocol;

stores the first virtual machine instructions in the second memory device, wherein the second memory device contains only one first virtual machine instruction at any time;

receives first template state data to initialize the first finite state machine; and executes the first virtual machine instructions using the first template state data and the first finite state machine.

2. (Currently Amended) The telephone gateway device ~~telecommunication protocol engine device~~ of claim 1, wherein the first telecommunication protocol includes at least one of a Session Initiation Protocol (SIP), a H.323 protocol, a STUN protocol, and a Dynamic Host Configuration Protocol (DHCP).

3. (Canceled).

4. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 1, further comprising an I/O port, and wherein the processor receives the first template state data via the I/O port.

5. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 1, wherein the first memory further includes second virtual machine instructions selected from a second telecommunication protocol template to implement a second telecommunication protocol.

6. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 5, wherein the second telecommunication protocol includes at least one of a Session Initiation Protocol (SIP), a H.323 protocol, a STUN protocol, and a Dynamic Host Configuration Protocol (DHCP).

7. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 5, wherein the second telecommunication protocol template includes one or more second template finite state machines.

8 -10. (Canceled).

11. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 1, wherein the processor is linked to the first memory device via a data bus.

12. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 11, wherein the data bus is a serial bus.

13. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 11, wherein the data bus is a parallel bus.

14. (Currently Amended) A method of implementing a telecommunication protocol, the method comprising:

receiving, at a telephone gateway device, a call at a processor to implement a first telecommunication protocol;

selecting a first telecommunication protocol template, wherein the first telecommunication protocol template includes one or more virtual machine instructions to implement the first telecommunication protocol;

causing a virtual machine to read a first virtual machine instruction of the first telecommunication protocol template from a first memory device;

initializing a first finite state machine defined by the first telecommunication protocol template using first template state data to arrive at a first template state; and

sending a message to a called party;

when the called party does not respond to the message:

executing a play fast busy tone command, and

entering a fast busy tone state;

when a response from the called party indicates that the called party has answered:

executing an init vocoder instruction,

executing a send instruction, and

entering a voice state; and

storing updated first template state data in a second memory device, wherein the updated first template state data is determined based on the first finite state machine.

15. (Currently Amended) The method of claim 14, wherein the first telecommunication protocol includes one of a Session Initiation Protocol (SIP), a H.323 protocol, ~~and~~ a STUN protocol, and a dynamic host configuration protocol (DHCP).

16. (Canceled).

17. (Previously Presented) The method of claim 14, wherein the first template state data is received by a telecommunication protocol engine.

18. (Currently Amended) The method of claim 14, ~~wherein the method further~~
comprising comprises:
receiving a call at the processor to implement a second telecommunication protocol;
selecting a second telecommunication protocol template, wherein the second
telecommunication protocol template includes one or more virtual machine
instructions to implement the second telecommunication protocol;
causing the virtual machine to read a first virtual machine instruction of the second
telecommunication protocol template from the first memory device;
initializing a second finite state machine defined by the second telecommunication
protocol template using second template state data to arrive at a second template
state; and
storing updated second template state data in the second memory device, wherein the
updated second template state data is determined based on the second finite state
machine.

19. (Currently Amended) The method of claim 18, wherein the second
telecommunication protocol includes one of a Session Initiation Protocol (SIP), a H.323
protocol, ~~and~~ a STUN protocol, and a dynamic host configuration protocol (DHCP).

20. (Canceled).

21. (Currently Amended) The method of claim 18, wherein the second template state data
is received by a telecommunication protocol engine.

22. (Previously Presented) The method of claim 18, wherein the first template state data
is determined based on the second finite state machine.

23. (Previously Presented) The method of claim 18, wherein the second template state
data is determined based on the first finite state machine.

24. – 43. (Canceled).

44. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 1, wherein execution of the first virtual machine instructions changes the first template state data based on the first finite state machine, and wherein the processor stores the changed first template state data at the second memory device.

45. (Canceled).

46. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 44, wherein the second memory device does not store an entire set of virtual machine instructions from the first telecommunication protocol template at a single time.

47. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 1, wherein the first memory device has a larger storage capacity than the second memory device.

48. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 1, wherein the first memory device ~~includes~~ is a FLASH memory device.

49. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 48, wherein the second memory device ~~includes~~ is a RAM memory device.

50. (Currently Amended) The telephone gateway device telecommunication protocol engine device of claim 5, wherein the processor:

causes the virtual machine to read the second virtual machine instructions from the first memory device in response to a request to implement the second telecommunication protocol;

stores the second virtual machine instructions in the second memory device;

receives second template state data related to initializing a second finite state machine related to the second telecommunication protocol; and

executes the second virtual machine instructions using the second template state data and the second finite state machine.

51. (Canceled).

52. (Canceled).

53. (Currently Amended) The method of claim [52] 14, wherein the second memory device contains only one first virtual machine instruction at any time ~~wherein no more than one virtual machine instruction to implement the first telecommunication protocol is stored in the second memory at any time.~~

54. (Currently Amended) The method of claim 14, further comprising:

causing the virtual machine to read a second virtual machine instruction of the first telecommunication protocol template from the first memory device based on the updated first template state data and the first finite state machine;

using the first finite state machine to determine a second template state data; and

storing the second template state data in the second memory device, wherein the updated first template state data is determined based on the first finite state machine.

55. (Currently Amended) The method of claim 54, wherein the first virtual machine instruction and the second virtual machine instruction are not stored in the second memory device concurrently.

56. (New) A telephone gateway device comprising:

- a first memory device storing one or more telecommunication protocol templates, each of the one or more telecommunication protocol templates including a plurality of virtual machine instructions to implement a telecommunication protocol and to implement a finite state machine;
- a second memory device to store state data related to a first finite state machine;...
- a processor having access to the first memory device, the second memory device the processor configured to:
 - read first virtual machine instructions from the first memory device, the first virtual machine instructions selected from a first telecommunication protocol template to implement a first telecommunication protocol;
 - store the first virtual machine instructions in the second memory device, wherein the second memory device contains only one first virtual machine instruction at any time;
 - receive first template state data to initialize the first finite state machine; and
 - execute the first virtual machine instructions using the first template state data and the first finite state machine.